

# How to give executable permission to all Python scripts in Linux?

Asked 6 years ago   Modified 1 year, 9 months ago   Viewed 10k times

---



Suppose I have a python script called **a.py** like this:

4



```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
# Author    : Bhishan Poudel
# Date      : Jul 13, 2016
```

```
# Imports
```

```
# Script
print("hello")
```

I can run this scripts in two ways:

**Using python interpreter:**

```
python3 a.py
```

**Changing permission**

```
chmod a+x a.py; ./a.py
```

## QUESTION

How can I run any new or old python scripts without using `chmod a+x script_name` all the times.

I have root access and user access both to my computer.

Basically i want executable permission to all the .py files, how can we do so?

I tried different shebangs such as:

```
#!/usr/bin/python3
#!/usr/bin/env python3
#!/usr/local/bin/python3
#!/usr/local/bin/env python3
```

The python interpreter is also in the \$PATH. The output of echo \$PATH is following:

```
/Library/Frameworks/Python.framework/Versions/2.7/bin:/Library/Frameworks
/Python.framework/Versions/3.5/bin:/usr/bin:/bin:/usr/sbin:/sbin:/usr/local
/bin:/opt/X11/bin:/usr/texbin:/Library/Frameworks/Python.framework/Versions
/2.7/bin:/Library/Frameworks/Python.framework/Versions/3.5/bin:/opt/local
/bin:/Users/poudel/phosim:/Users/poudel/Applications:/usr/local/octave/3.8.0
/bin:/Users/poudel/Applications/Geany.app/Contents/MacOS:/opt/local/bin:/Users
/poudel/phosim:/Users/poudel/Applications:/usr/local/octave/3.8.0
/bin:/Applications/Geany.app/Contents/MacOS:/opt/local/bin:/Users/poudel
/phosim:/Users/poudel/Applications:/usr/local/octave/3.8.0/bin:/Applications
/Geany.app/Contents/MacOS/
```

Also, ls /usr/bin/py\* has:

```
/usr/bin/pydoc*          /usr/bin/python2.5@      /usr/bin/pythonw*
/usr/bin/pydoc2.5@      /usr/bin/python2.5-config@ /usr/bin/pythonw2.5@
/usr/bin/pydoc2.6@      /usr/bin/python2.6@      /usr/bin/pythonw2.6@
/usr/bin/pydoc2.7@      /usr/bin/python2.6-config@ /usr/bin/pythonw2.7@
/usr/bin/python*        /usr/bin/python2.7@
/usr/bin/python-config*  /usr/bin/python2.7-config@
```

### Related links:

<http://effbot.org/pyfaq/how-do-i-make-a-python-script-executable-on-unix.htm>

[Permission Denied when executing python file in linux](#)

[bash permission denied for python](#)

[Permission denied when launch python script via bash](#)

[python](#) [bash](#) [path](#) [executable](#) [chmod](#)

Share Follow

edited Oct 3, 2019 at 9:12



jww

91.7k

84

376

822

asked Jul 13, 2016 at 15:23



BhishanPoudel

1

---


You probably *don't* want all `.py` files to be executable; modules that aren't intended to be run as scripts should not have their executable bit set (or at least, there is no reason to set it). – [chepner](#) Jul 13, 2016 at 15:40

---

@chepner, I appreciate your idea, but i wanted to run files without using `chmod a+x script_name.py` all the time for all the files. For future reference, I will consider your idea too. Thanks. – [BhishanPoudel](#) Jul 13, 2016 at 15:53

---

## 4 Answers

Sorted by: Trending sort available ⓘ Highest score (default) 



## The hard way

6

Run below with root privilege:



```
find /your/path/ -type f -name "*.py" -exec chmod u+x {} \;
```



### Note:



chmod need not be run as root if you're the owner of .py file.

## The smart way

Write a script to take care of this.

```
#!/bin/bash
if [ -f "$1" ]
then
geany "$1" # You could also use xdg-open if you set geany to open .py files
else
cp /path/to/python/startup/template "$1" # You may omit this if you don't have a
default template
chmod u+x "$1"
geany "$1"
fi
```

Save the script as, say, pycreator in say /usr/bin/ , then do

```
chown root:root /usr/bin/pycreator
chmod +x-w /usr/bin/pycreator
```

To create a new script using pycreator , do

```
pycreator calculator.py
```

Also [\[this\]](#) answer pointed to by [@choroba](#) in his comment provides valuable insight in this regard.

Share Follow

edited May 23, 2017 at 11:46

answered Jul 13, 2016 at 15:25



Community Bot

1 1



sjsam

20.8k 4 49 96

3 Also, configure your editor to add the execution flag for future .py files created. – [choroba](#) Jul 13, 2016 at 15:31

@choroba : I like this answer, but is it possible to configure an editor say `vi` to automatically add `+x` while saving a file with particular extension say `.sh` ? – [Mona\\_Cax](#) Jul 13, 2016 at 16:40

Using the idea of [@sjsam](#), I did following:

3 Suppose I have a file `hello.py` in any location.

`cd` to that location

```
find $PWD -type f -name "*.py" -exec chmod u+x {} \;  
./hello.py
```

# Now, i can create any number of .py files in that folder and run `./filename`

# Note: if we are running as user permission, and also have `sudo` access,  
we can also `do`:  
`sudo -H find $PWD -type f -name "*.py" -exec chmod u+x {} \;`

We should not use `sudo` unless absolutely necessary.

Thanks to [sjsam](#).

Share Follow

edited Jul 13, 2016 at 15:54

answered Jul 13, 2016 at 15:39



BhishanPoudel

1

▲ Just type this command in terminal.

0 sudo su



Share Follow

answered Nov 18, 2017 at 6:06



[jakir hussain](#)

306 2 18

▲ Try this **chmod +x \*.py** it works on my PC(OS:Ubuntu 20.4), Also i am using **#!/usr/bin/env python3** shebang

0 Share Follow



answered Oct 10, 2020 at 6:42



[WasitShafi](#)

467 1 5 10